is no published work available on how to tag Telugu corpus using this BIS-POS tagset. Hence, the paper describes how Telugu corpus is (to be) tagged and issues encountered while tagging the corpus. The main aim of the paper is to develope a manual which is to be used for human-annotation tasks and for building automatic systems.

Augmenting Recurrent Neural Network Language Models with Subword Information

Lyan Verwimp, Joris Pelemans, Hugo Van Hamme, Patrick Wambacq

Many tasks in natural language processing (e.g. speech recognition, machine translation, ...) require a language model: a model that predicts the next word given the history. Traditionally, these models are count-based and assign a probability to a sequence of n words based on the frequency of that sequence in the training text. However, these so-called n-gram models suffer from data sparsity and are not capable of modeling long-distance dependencies. Neural network-based language models (partly) solve data sparsity issues by projecting words onto a continuous space such that better generalizations can be made. Moreover, recurrent neural network language models (RNNLMs) are capable of modeling long-distance dependencies because they have a memory. RNNLMs take as input a one-hot encoded vector of the current word together with a copy of the hidden layer at the previous time step, which is then projected onto a hidden layer, from which a probability distribution for the next word is computed in the output layer. As a result, words that occur in similar contexts end up close to each other in the continuous space. However, since RNNLMs treat words as atoms, it is not possible to capture the formal/morphological properties of words.

In the context of the project STON (IWT - INNOVATIEF AANBESTEDEN), we explore the addition of subword information to the input of RNNLMs. In this way, we transform the projection such that not only words occurring in similar contexts but also words with a comparable structure get vector representations that are close to each other.

Automatic detection and correction of preposition errors in learners' Dutch

Lennart Kloppenburg, Malvina Nissim

In this work we address the automatic detection and correction of preposition errors in essays written in L2 Dutch by leveraging native data. Using Support Vector Machines on the Lassy Large corpus, which is supposed to exhibit correct preposition usage, we created language models for the 15 most frequent prepositions in Dutch using 20M sentences (a multiclass model of preposition selection), and for preposition presence or absence, using 2M sentences (a binary model of preposition detection). For both models, we used a set of features based on token and POS n-grams and dependency relations obtained via the Alpino parser. The binary model predicts if a context vector has a preposition label or not. The multiclass model predicts a specific preposition given a context vector. These models form a pipeline for detecting and correcting preposition errors, of the following three types: 1. Insertion (preposition invoked erroneously) 2. Deletion (preposition omitted erroneously) 3. Substitution (preposition picked erroneously) The models were evaluated on native and learners' data. On L1 data, the binary and selection models score respective F-scores of 100% and 75%. Because the learner corpus (Leerdercorpus Nederlands) is not error-annotated, we used crowdsourcing to evaluate performance. We gathered human judgements for 1,499 cases and compared the system's decisions on them with the annotators' choices. Of all substitution errors identfied